

Exhibit B
'tzlbac14.log'

```

SQL> @tzlas01
SQL>
SQL> CONNECT LBACSYS/LBACSYS
Connected.
SQL>
SQL> -- Create two SA policies
SQL> EXECUTE SA_SYSDBA.CREATE_POLICY('SA1','SA1_COL','ALL_CONTROL');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_SYSDBA.CREATE_POLICY('SA2','SA2_COL','NO_CONTROL');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Initialize PUBLIC labels for them
SQL> EXECUTE SA_LABELS.CREATE_LEVEL('SA1',0,'PUBLIC','PUBLIC Level');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_LEVEL('SA2',0,'PUBLIC','PUBLIC Level');

PL/SQL procedure successfully completed.

SQL>
SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1',10,'public');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa2',10,'public');
BEGIN SA_LABEL_ADMIN.CREATE_LABEL('sa2',10,'public'); END;

*
ERROR at line 1:
ORA-12432: LBAC error: Label with the given label_tag: 10 already exists
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_LABEL_ADMIN", line 57
ORA-06512: at line 1

SQL>
SQL> -- Setup some labels for policy SA1
SQL> EXECUTE SA_LABELS.CREATE_LEVEL('sa1',10,'c','confidential');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_LEVEL('sa1',20,'s','SECRET');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_LEVEL('sa1',30,'ts','Top Secret');

PL/SQL procedure successfully completed.

SQL>
SQL> EXECUTE SA_LABELS.CREATE_COMPARTMENT ('sa1', 5, 'A', 'ALPHA');

```

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_COMPARTMENT ('sa1', 10, 'b', 'beta');

PL/SQL procedure successfully completed.

SQL>

SQL> EXECUTE SA_LABELS.CREATE_GROUP ('sa1', 5, 'G1','group 1');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_GROUP ('sa1', 51, 'G2','group 2','G1');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABELS.CREATE_GROUP ('sa1', 52, 'G3','group 3','G1');

PL/SQL procedure successfully completed.

SQL>

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1', 200,'c');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1', 225,'c:b,a');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1',210,'c:a');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1',205,'c::g2');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1', 300,'s');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('sa1', 310,'s:a');

PL/SQL procedure successfully completed.

SQL>

SQL> -- Generate some labels

SQL> SELECT LABEL_TO_CHAR(TO_SA_LABEL('sa1','c:a:g1')) FROM DUAL;

LABEL_TO_CHAR(TO_SA_LABEL('SA1','C:A:G1'))

C:A:G1

1 row selected.

SQL> SELECT LABEL_TO_CHAR(TO_SA_LABEL('sa1','s:a,b')) FROM DUAL;

LABEL_TO_CHAR(TO_SA_LABEL('SA1','S:A,B'))

S:A,B

1 row selected.

SQL> SELECT LABEL_TO_CHAR(TO_SA_LABEL('sa1','public:a:g1')) FROM DUAL;

LABEL_TO_CHAR(TO_SA_LABEL('SA1','PUBLIC:A:G1'))

PUBLIC:A:G1

1 row selected.

SQL>

SQL> COL POLICY_NAME FORMAT A15

SQL> COL LABEL FORMAT A20

SQL> SELECT * FROM DBA_SA_LABELS;

POLICY_NAME	LABEL	LABEL_TAG	LABEL_TYPE
SA1	PUBLIC	10	USER LABEL
SA1	C	200	USER/DATA LABEL
SA1	C::G2	205	USER/DATA LABEL
SA1	C:A	210	USER/DATA LABEL
SA1	C:A,B	225	USER/DATA LABEL
SA1	S	300	USER/DATA LABEL
SA1	S:A	310	USER/DATA LABEL
SA1	C:A:G1	1000000000	USER/DATA LABEL
SA1	S:A,B	1000000001	USER/DATA LABEL
SA1	PUBLIC:A:G1	1000000002	USER/DATA LABEL

10 rows selected.

SQL>

SQL> col labelvalue format a20

SQL> col policy_name format a10

SQL> SELECT * from dba_sa_labels;

POLICY_NAM	LABEL	LABEL_TAG	LABEL_TYPE
SA1	PUBLIC	10	USER LABEL
SA1	C	200	USER/DATA LABEL
SA1	C::G2	205	USER/DATA LABEL
SA1	C:A	210	USER/DATA LABEL
SA1	C:A,B	225	USER/DATA LABEL
SA1	S	300	USER/DATA LABEL
SA1	S:A	310	USER/DATA LABEL
SA1	C:A:G1	1000000000	USER/DATA LABEL
SA1	S:A,B	1000000001	USER/DATA LABEL
SA1	PUBLIC:A:G1	1000000002	USER/DATA LABEL

10 rows selected.

SQL>

SQL> -- Set user labels

SQL> EXECUTE SA_USER_ADMIN.SET_LEVELS('sa1','scott','s','c');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_USER_ADMIN.SET_COMPARTMENTS('sa1','scott','a,b');

PL/SQL procedure successfully completed.

SQL> EXECUTE SA_USER_ADMIN.SET_GROUPS('sa1','scott','G1');

PL/SQL procedure successfully completed.

SQL> SELECT * FROM dba_sa_user_levels ORDER BY policy_name, user_name;

POLICY_NAM	USER_NAME	MAX_LEVEL
MIN_LEVEL	DEF_LEVEL	
ROW_LEVEL		
SA1	SCOTT	S
C		S
S		

1 row selected.

SQL> SELECT * FROM dba_sa_user_compartments ORDER BY policy_name, user_name;

POLICY_NAM	USER_NAME	COMP	RW_AC	D
SA1	SCOTT	A	WRITE	Y
SA1	SCOTT	B	WRITE	Y

2 rows selected.

SQL> SELECT * FROM dba_sa_user_groups ORDER BY policy_name, user_name;

POLICY_NAM	USER_NAME	GRP	RW_AC	D
SA1	SCOTT	G1	WRITE	Y

1 row selected.

SQL>

SQL> -- Look at session labels

SQL> CONNECT scott/tiger

Connected.

SQL>

```

SQL> create or replace FUNCTION get_list (pol IN VARCHAR2)
  2  RETURN VARCHAR2 IS
  3      test_list lbacsys.lbac_label_list;
  4  begin
  5      test_list:=lbac_session.effective_labels(pol);
  6      RETURN label_list_to_named_char(test_list,'effective');
  7  END;
  8  /

```

Function created.

```

SQL>
SQL> select get_list('sa1') from dual;

```

GET_LIST('SA1')

```

-----
MAX READ LABEL='S:A,B:G1,G2,G3',MAX WRITE LABEL='S:A,B:G1,G2,G3',MIN WRITE LABEL
='C',READ LABEL='S:A,B:G1,G2,G3',WRITE LABEL='S:A,B:G1,G2,G3',ROW LABEL='S:A,B:G
1,G2,G3'

```

1 row selected.

```

SQL> select get_list('sa2') from dual;

```

GET_LIST('SA2')

1 row selected.

```

SQL>
SQL> SQL>

```